Homework 12

Christophe Hunt

April 27, 2017

Contents

1 Assignment Introduction

Using the $stats$ and $boot$ libraries in R perform a cross-validation experiment to observe the bias variance tradeoff. You'll use the auto data set from previous assignments. This dataset has 392 observations across 5 variables. We want to fit a polynomial model of various degrees using the glm function in R and then measure the cross validation error using cv.glm function.

Fit various polynomial models to compute mpg as a function of the other four variables $acceleration, weight, horsepower, anddi$ using glm function. For example:

glm.fit=glm(mpg~poly(disp+hp+wt+acc,2), data=auto)
cv.err5[2]=cv.glm(auto,glm.fit,K=5)$delta[1]

will fit a 2nd degree polynomial function between mpg and the remaining 4 variables and perform 5 iterations of cross-validations. This result will be stored in a $cv.err5$ array. $cv.glm$ returns the estimated cross validation error and its adjusted value in a variable called delta. Please see the help on $cv.glm$ to see more information.

Once you have fit the various polynomials from degree 1 to 8, you can plot the cross validation error function as

degree=1:8
plot(degree,cv.err5,type='b')

For you assignment, please create an R-markdown document where you load the auto data set, perform the polynomial fit and then plot the resulting 5 fold cross validation curve.

2 Exercise

Load in the auto-data

```
library(tidyverse)
df <- as_tibble(read.table(paste0("https://raw.githubusercontent.com",
                                  "/ChristopheHunt/MSDA---Coursework",
                                  "/master/Data%20605/Assignment%2011/",
                                  "auto-mpg.data")))
colnames(df) <- c("displacement", "horsepower", "weight", "acceleration", "mpg")
```

We modify the provided function and create the for loop for 1 to 8 degree polynomial models.

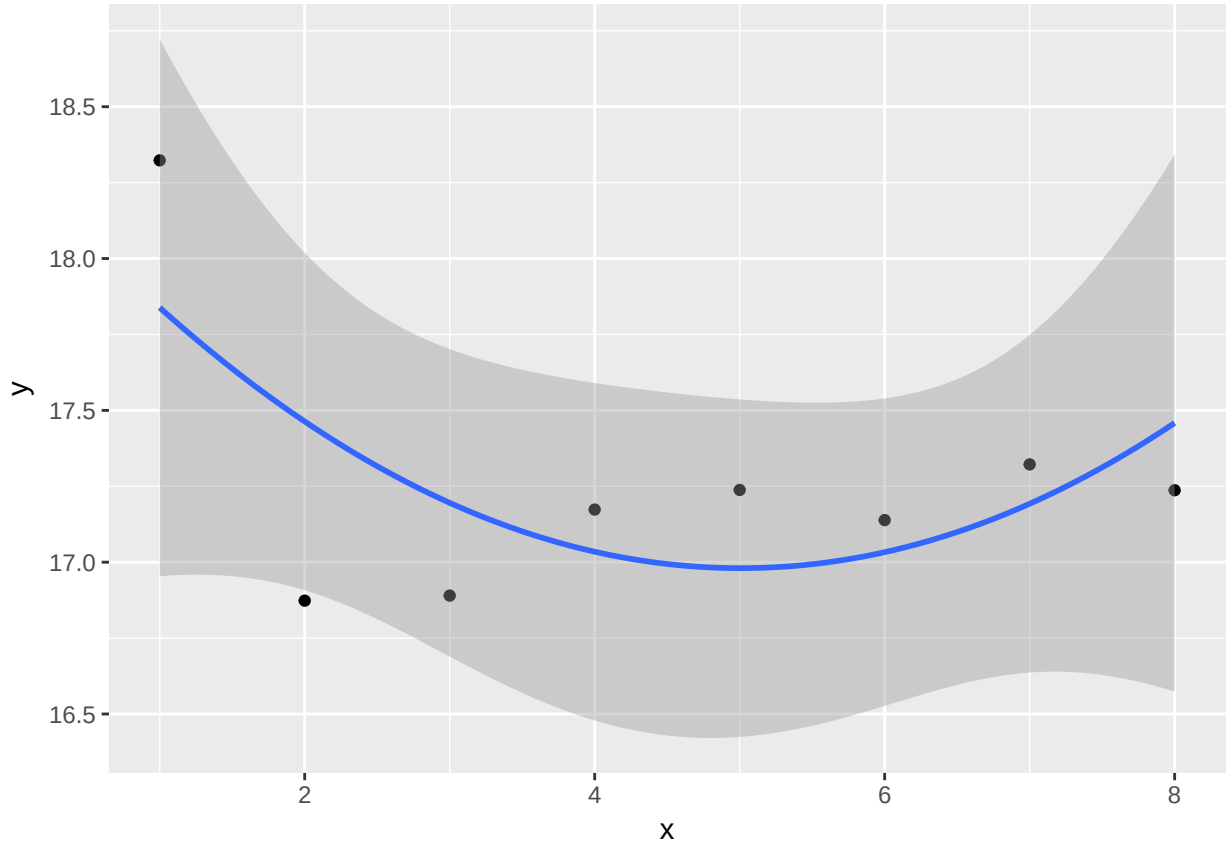
```
library(stats)
library(boot)
cv.err5 <- list()

for (i in 1:8){
glm.fit <- glm(mpg~poly(displacement+horsepower+weight+acceleration, i), data = df)

cv.err5[[i]] <- list(x = cv.glm(df, glm.fit, K = 5)$delta[1], y = i)
}
```

The next chunk is to return the list to a x,y data frame. I think the above loop could be improved to return a x,y data frame.

```
plot_df <- NULL
for (i in 1:8){
plot_df <- rbind(plot_df,(unlist(cv.err5[[i]])))
}
colnames(plot_df) <- c("y", "x")
```

Plotting the data and adding a smoothing line to illustrate the expected u shape illustrating bias and variance.

```
ggplot(data = as.data.frame(plot_df), aes(x=x, y=y)) +
  geom_point()+
  stat_smooth(span = 100, method = "loess")
```



Your output should show the characteristic U-shape illustrating the tradeoff between bias and variance.